Index.js  
import React from "react";

import ReactDOM from "react-dom";

import "./styles.css";

import App from "./App";

ReactDOM.render(

<React.StrictMode>

<App />

</React.StrictMode>,

document.getElementById("root")

);

App.js  
  
import React, { useEffect, useState } from "react";

function App() {

// State variables

const [products, setProducts] = useState([]);

const [loading, setLoading] = useState(true);

const [error, setError] = useState(null);

// Fetch products on component mount

useEffect(() => {

const fetchProducts = async () => {

try {

const response = await fetch("https://fakestoreapi.com/products");

if (!response.ok) {

throw new Error("Network response was not ok");

}

const data = await response.json();

setProducts(data);

} catch (error) {

setError(error.message);

} finally {

setLoading(false);

}

};

fetchProducts();

}, []);

// Display loading state

if (loading) {

return <div className="loading">Loading...</div>;

}

// Display error message if any

if (error) {

return <div className="error-message">Error: {error}</div>;

}

return (

<div className="product-listing">

<h1>Product Listing</h1>

<div className="products">

{products.map((product) => (

<div key={product.id} className="product-card">

<img

src={product.image}

alt={product.title}

style={{ width: "150px", height: "150px" }}

/>

<h2>{product.title}</h2>

<p>${product.price.toFixed(2)}</p>

<p className="category">{product.category}</p>

<button>Add to Cart</button>

</div>

))}

</div>

</div>

);

}

export default App;  
  
styles.css  
  
/\* General body and layout styles \*/

body {

font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;

background-color: #f5f5f5;

margin: 0;

padding: 0;

}

h1 {

color: #333;

text-align: center;

margin-bottom: 30px;

font-size: 2.5rem;

}

.product-listing {

max-width: 1000px;

margin: 0 auto;

padding: 20px;

background-color: #fff;

border-radius: 10px;

box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);

}

.products {

display: grid;

grid-template-columns: repeat(auto-fill, minmax(250px, 1fr));

gap: 20px;

}

.product-card {

border: 1px solid #ddd;

border-radius: 10px;

padding: 20px;

text-align: center;

box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

background-color: #fff;

transition: transform 0.3s ease, box-shadow 0.3s ease;

}

.product-card:hover {

transform: scale(1.05);

box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);

}

.product-card img {

border-radius: 10px;

margin-bottom: 15px;

}

.product-card h2 {

font-size: 1.25rem;

color: #333;

margin: 10px 0;

}

.product-card p {

font-size: 1rem;

color: #666;

margin: 5px 0;

}

.product-card .category {

font-style: italic;

color: #777;

margin-top: 10px;

}

button {

padding: 10px 20px;

font-size: 1rem;

color: #fff;

background-color: #007bff;

border: none;

border-radius: 5px;

cursor: pointer;

transition: background-color 0.3s ease;

}

button:hover {

background-color: #0056b3;

}

.loading,

.error-message {

font-size: 1.25rem;

color: #555;

text-align: center;

margin-top: 50px;

}

/\* Responsive design for smaller devices \*/

@media (max-width: 768px) {

.product-card img {

width: 100px;

height: 100px;

}

h1 {

font-size: 2rem;

}

.product-card h2 {

font-size: 1.1rem;

}

}